





Robust surface-correlation-function evaluation from experimental discrete digital imagesAleksei Samarin,^{1,2} Vasily Postnicov ¹, Marina V. Karsanina,¹ Efim V. Lavrukhin,^{1,2} Dina Gafurova ³,
Nikolay M. Evstigneev ⁴, Aleksey Khlyupin,¹ and Kirill M. Gerke ^{1,*}¹*Schmidt Institute of Physics of the Earth of Russian Academy of Sciences, Moscow 107031, Russia*²*Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow 119991, Russia*³*Oil and Gas Research Institute Russian Academy of Sciences (OGRI RAS) 3, Gubkina Street, Moscow 119333, Russian Federation*⁴*Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences, Moscow 117312, Russia*

(Received 23 August 2022; revised 4 February 2023; accepted 18 April 2023; published 26 June 2023)

Correlation functions (CFs) are universal structural descriptors; surface-surface F_{ss} and surface-void F_{sv} CFs are a subset containing additional information about the interface between the phases. The description of the interface between pores and solids in porous media is of particular importance and recently Ma and Torquato [Phys. Rev. E **98**, 013307 (2018)] proposed an elegant way to compute these functions for a wide variety of cases. However, their “continuous” approach is not always applicable to digital experimental 2D and 3D images of porous media as obtained using x-ray tomography or scanning electron microscopy due to nonsingularities in chemical composition or local solid material’s density and partial volume effects. In this paper we propose to use edge-detecting filters to compute surface CFs in the “digital” fashion directly in the images. Computed this way, surface correlation functions are the same as analytically known for Poisson disks in case the resolution of the image is adequate. Based on the multiscale image analysis we developed a $C_{0.5}$ criterion that can predict if the imaging resolution is enough to make an accurate evaluation of the surface CFs. We also showed that in cases when the input image contains all major features, but do not pass the $C_{0.5}$ criterion, it is possible with the help of image magnification to sample CFs almost similar to those obtained for high-resolution image of the same structure with high $C_{0.5}$. The computational framework as developed here is open source and available within the `CorrelationFunctions.jl` package developed by our group. Our “digital” approach was applied to a wide variety of real porous media images of different quality. We discuss critical aspects of surface correlation functions computations as related to different applications. The developed methodology allows applying surface CFs to describe the structure of porous materials based on their experimental images and enhance stochastic reconstructions or super-resolution procedures, or serve as an efficient metrics in machine learning applications due to computationally effective GPU implementation.

DOI: [10.1103/PhysRevE.107.065306](https://doi.org/10.1103/PhysRevE.107.065306)**I. INTRODUCTION**

Correlation functions (CFs) are invaluable universal descriptors of structure and in this context are utilized in a multitude of scientific disciplines: material sciences [1,2], rock [3] and soil physics [4–6], cosmology [7], and food engineering [8], to name just a handful. As such, CFs were used to characterize the morphology [9] and representativeness via correlation lengths [10,11], compare structures [12–14], compress structural information [6,15,16], describe structural dynamics [5,17,18], extract features for deep learning [6,19–21], perform stochastic reconstructions [22–26], and fuse multiscale images [15,27,28]. Stochastic reconstruction is a special topic of interest, as this approach allows to solve an inverse problem and recover structure from a known set of correlation functions—and this ability for recovery is the basis for majority of potential usages in the list above. Early reconstruction techniques mainly involved two-point probability S_2 function, but were improved to include lineal L_2 function [10,23,29], cluster C_2 function [12,30] and poten-

tially surface F_{SS} and F_{SV} functions. With the proper handling [24] it is possible to incorporate numerous CFs into a reconstruction procedure. Increasing the number (and order) of functions leads to higher information content of the CFs set [31–33] and, thus, allows reconstructing and performing stationarity/representativeness characterization [14] for structures of any complexity.

There are two ways to obtain correlation functions for a given structure at hand—either measure them experimentally with the help of scattering intensity [34,35], or measured from images. The first approach suffers from the limitation of CFs that can be obtained this way [36], while the second one provides information with limited resolution or/and resolution to field-of-view ratio [15]. Moreover, the most useful imaging methods such as x-ray computed tomography (XCT) and scanning electron microscopy (SEM) due to their underlying physical principles provide gray-scale images which, as we shall argue later on, need to be segmented into constituent phases before CFs computation. It is important to note that such gray-scale images do not represent spatial distribution of the phases, but, for example, x-ray spatial attenuation or electron back-scattering intensities. However, if properly segmented, high-resolution digital images do provide a pos-

*kg@ifz.ru

sibility to compute any correlation function. Experimental measurements using small angle scattering (SAS) are traditionally used to evaluate S_2 , but it is also possible to relate scattering intensities to surface correlation functions [37,38]. It would be of great practical importance to obtain surface correlation functions from both SAS and imaging (focused ion beam milling combined with SEM allows to obtain the finest imaging resolution and sample CFs from polished surfaces as opposed to surface imaging [39]) to estimate coefficients for surface and bulk scattering (see Eq. (13) in Ref. [38]). But there is a catch—SAS possesses close to infinite surface resolution as opposed to digital pixelized images with inherent segmentation problems due to partial volume effects, something we discuss extensively in this paper.

Recently, Ma and Torquato [38] laid foundation to precise surface correlation function computations and showed the usefulness of these CFs evaluation for numerous problems. They have implemented an elegant solution with the help of infinite resolution random fields that (if the fields are thresholded) allows finding an exact intersection with the sampling line. In addition, they also showed that surface CFs for digital images can be computed by converting integer fields (i.e., location of the phases) to float fields with the help of Gaussian filter [38]. Unfortunately, their solution is not readily applicable to the majority of grayscale images due to nonsingularity in their chemical constitution (pixels/voxels in studied grayscale images may contain multiple material phases). The reason is the difference between the contour and real interface between different phases within the material due to the partial volume effects [40] for XCT imaging—the presence of multiple phases within the same pixel/voxel. In other words, the attenuation is a function of both density and atomic number, which usually are distributed nonuniformly below the XCT imaging resolution. Somewhat similar is also relevant for SEM imaging, as secondary or back-scattered electrons are effectively a convolution of partial signals coming from different depths [41]. The exact subvoxel thresholding based on gray-scale image (similar to the technique used in [38]) is only available in case it is monomineral, i.e., the solid phase consists of a completely chemically homogeneous substance that contrasts perfectly with air/vacuum filled pore phase—this is rarely the case for natural materials. Moreover, even if the image of a monomineral sample (in natural porous media carbonate rocks can be considered monomineral depending on their genesis) has subresolution features on XCT image, while it is possible to compute the porosity of the voxel, it is still not possible to extract the interface. All these and additional problems (such as, for example, experimental noise and artifacts from inverse Radon transform) arising during gray-scale image processing as related to image segmentation were extensively discussed elsewhere [42]. Thus, we argue, that the usage of current state-of-the-art segmentation techniques is necessary prior to the conversion to a coarse-grained scalar field. Moreover, finding an intersection between a line segment and the contour is a computationally expensive procedure that prevents on-the-fly application of surface functions. If a binary image (consisting of pores and solids) is produced by adequate segmentation technique, then we still lack a robust and computationally effective procedure to evaluate surface CFs from experimental XCT and SEM

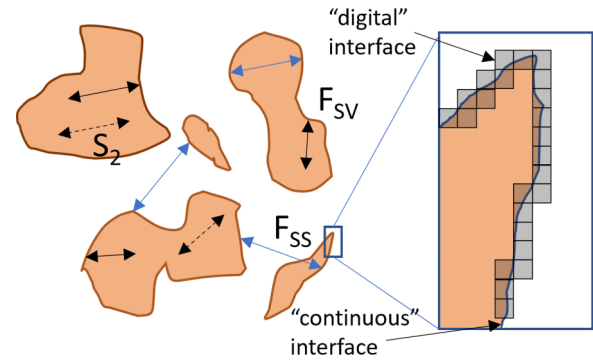


FIG. 1. A schematic depiction of a binary porous media (pores are shown in color) with examples of positive events for surface F_{SS} , F_{SV} and two-point probability S_2 correlation functions. The zoomed in area represents the difference between the true “continuous” interface in between pore and solid phases with pixelized “digital” interface emerging due to limited resolution of digital images.

images of natural heterogeneous materials, for instance, rocks and soils.

To compute surface functions one first needs to elucidate the interface between two phases (we shall consider only pores and solids, but, obviously, the computations can be performed to multiphase systems in the similar manner). The interface area has infinitesimal volume, but its location on the digital image is not easy due to the pixelization of the interface between the binary phases [38]. By adopting the interface width $\varepsilon = \text{pixel/voxel size}$ one can create a very crude approximation of the real interface, but the usage of interfaces between voxels “as is” leads to known problems in surface area and surface geometry evaluation, in application of 3D imaging for geometry and topology analysis [43,44] or energy minimization problems [45]. Another option would be to describe the boundaries between voxels with some curves, e.g., splines. This way it would be possible to perform surface CFs sampling by line intersection as described by Ma and Torquato [38]. While an exact boundary can be obtained for deterministic structures such as disk packings, splines would provide only a very approximate solution for the boundaries of arbitrary digitized structure due to the limit in resolution for a given image [46]. In other words, contours extracted from digital images will approach real boundaries between phases only when the spatial resolution approaches infinity. But the same is also true for digital pixelized/voxelized images. Thus, the options to compute surface correlation functions include “continuous” approach (such as implemented in [38]) and “digital” approach (similar to computation of S_2 from digital images) as depicted in Fig. 1. Here we propose a modified “digital” approach that lies in between the exact and pixelized solutions for a number of reasons:

(1) For a binarized XCT or SEM image the interpolation of the interface using “continuous” or “digital” in the limit of $\varepsilon \rightarrow 0$ would produce the same results. Thus, as it is not possible to obtain the exact continuous interface from such an image, using “digital” approach is natural as applied to digital pixel/voxel images.

(2) “Continuous” approach is expensive numerically, as it requires to find the intersections between a line and a curve.

(3) Digital approach allows utilizing linear scan CFs computation or the fast Fourier transform (FFT)—the latter is the fastest way to evaluate full CF maps (not for all functions thought) and is very efficient on modern GPUs.

In this paper we build upon foundational work of Ma and Torquato [38] and develop a robust and efficient approach to compute surface correlation functions from digital 2D and 3D images. The rest of the manuscript is organized as follows: in Sec. III we provide all methodological details for surface CFs computation including analytical solutions to verify the proposed methodology and describe an image library for extensive testing of our algorithms, Sec. IV presents all major results of surface functions evaluations. Our algorithm is compared against the original algorithm of Ma and Torquato in Sec. V. Section VI applies the developed tools to evaluate surface CFs for different real porous media images, including real XCT and SEM data. We discuss obtained results, including the effects of image scaling, and outline future uses of surface functions within Sec. VII. The paper concludes with a summary in Sec. VIII.

II. THEORETICAL BACKGROUND

A. Correlation functions and definitions

First, we introduce an indicator function $I^{(i)}(\mathbf{x})$ which marks pixels of 2D and voxels of 3D digitized images as belonging to the phase i or not. It can be defined as

$$I^{(i)}(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in V_i, \\ 0 & \text{otherwise,} \end{cases}$$

where $V_i \subset \mathbb{R}^n$ is the region occupied by phase i . For statistically homogeneous media the ensemble average of $I^{(i)}$ equals volume fraction of a given phase. For binary media the following equality holds:

$$\phi_{\text{void}} + \phi_{\text{solid}} = 1, \quad \phi_i = \langle I^{(i)}(\mathbf{x}) \rangle.$$

In a similar fashion we can define an interface indicator function $M(\mathbf{x})$ which provides interface area s if averaged over the whole image:

$$M(\mathbf{x}) = |\nabla I^{(\text{solid})}(\mathbf{x})| = |\nabla I^{(\text{void})}(\mathbf{x})|, \quad (1)$$

$$\langle M(\mathbf{x}) \rangle = s. \quad (2)$$

The simplest, yet foundational correlation function is the two-point probability function S_2 which is defined as a probability that the ends of a random line segment belong to the same phase:

$$S_2^{(i)}(\mathbf{x}_1, \mathbf{x}_2) = \langle I^{(i)}(\mathbf{x}_1)I^{(i)}(\mathbf{x}_2) \rangle. \quad (3)$$

This equation can be further simplified for statistically homogeneous media, as S_2 will depend only on the relative displacement \mathbf{r} :

$$S_2^{(i)}(\mathbf{x}_1, \mathbf{x}_2) = S_2^{(i)}(\mathbf{r}).$$

The value of $S_2^{(i)}(\mathbf{r})$ at $\mathbf{r} = 0$ is a fraction of phase i in a medium:

$$S_2^{(i)}(0) = \phi_i.$$

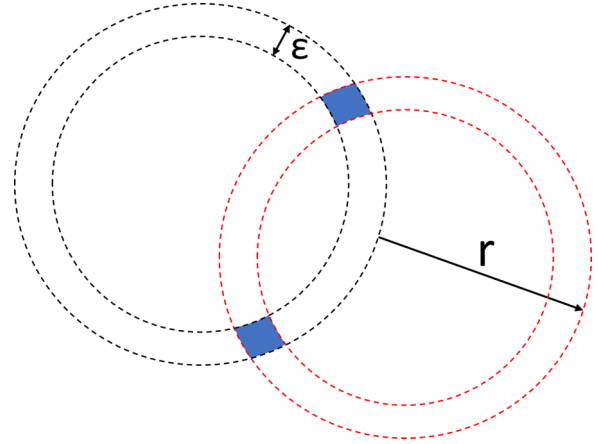


FIG. 2. Interpretation of F_{SS} as a self-intersection of the interface.

For isotropic media a vector displacement \mathbf{r} can be replaced with its length $r = |\mathbf{r}|$ and $S_2(\mathbf{r})$ with $S_2(r) = S_2(|\mathbf{r}|)$. Now, analogously to S_2 we can define surface-surface and surface-void correlation functions (for homogeneous and isotropic media straight away):

$$F_{ss}(r) = \langle M(x)M(x+r) \rangle, \quad (4)$$

$$F_{sv}(r) = \langle M(x)I^{(\text{void})}(x+r) \rangle. \quad (5)$$

The interface indicator function $M(x)$ is basically defined as infinity on the interface and zero elsewhere. Therefore, it is hard to directly compute $\langle M(x), M(x+r) \rangle$. We can replace the line of zero width and infinite value with a strip of width ε and value $\frac{1}{\varepsilon} M(x; \varepsilon)$. Then $F_{SS}(x; \varepsilon) = \langle M(x; \varepsilon), M(x+r; \varepsilon) \rangle$ is simply an intersection area of two strips—original and shifted—multiplied by $(\frac{1}{\varepsilon})^2$. And with $\varepsilon \rightarrow 0 \Rightarrow F_{SS}(r; \varepsilon) \rightarrow F_{SS}(r)$. Area between two black dashed circles on Fig. 2 represents $M(x; \varepsilon)$, red is for $M(x+r; \varepsilon)$; blue color marks the intersection area. The treatment of the interface in the case of F_{SV} is similar, but this function is not autocorrelation, but rather cross-correlation of the interface with the void phase. More detailed information on two-point probability and surface correlation functions can be found in the comprehensive Torquato's book [47].

B. Analytical solutions

For Poisson disks and balls one can derive exact analytical surface-surface and surface-void functions. For overlapping disks with radius R and centers generated by Poisson point process with parameter λ we have (see the derivation of these formulas in Appendix A):

$$F_{SV}(r) = S_2(r) \begin{cases} 2(\pi - B)R\lambda & r < 2R, \\ 2\pi R\lambda & \text{otherwise,} \end{cases} \quad (6)$$

$$F_{SS}(r) = S_2(r) \begin{cases} \frac{(2(B-\pi)R\lambda)^2 Ar + 4\sqrt{AR^2\lambda}}{Ar} & r < 2R, \\ (2\pi R\lambda)^2 & \text{otherwise,} \end{cases} \quad (7)$$

where S_2 is the regular two-point correlation function and

$$A = 4R^2 - r^2,$$

$$B = \arccos\left(\frac{r}{2R}\right).$$

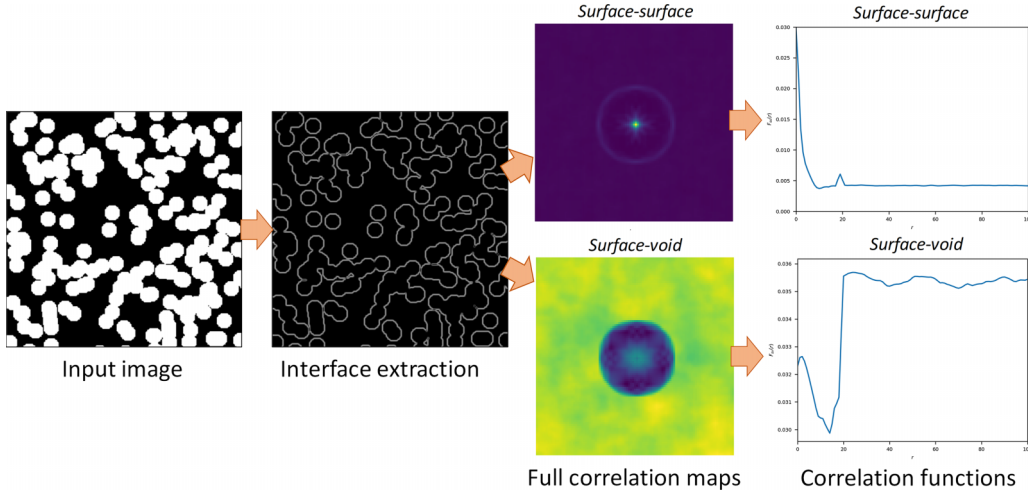


FIG. 3. Schematic example of the algorithm for surface functions computation (2D case is considered for visibility).

For 3D balls the relationships are readily available in the literature [38,47]:

$$F_{sv}(r) = S_2(r) \begin{cases} 4\pi R^2 \lambda \left(\frac{1}{2} + \frac{r}{4R}\right) & r < 2R, \\ 4\pi R^2 \lambda & \text{otherwise,} \end{cases}$$

$$F_{ss}(r) = S_2(r) \begin{cases} \left[4\pi R^2 \lambda \left(\frac{1}{2} + \frac{r}{4R}\right)\right]^2 + \frac{2\pi R^2 \lambda}{r} & r < 2R, \\ (4\pi R^2 \lambda)^2 & \text{otherwise.} \end{cases}$$

These analytical solutions will be used to verify the accuracy of our evaluation of surface CFs for such systems.

III. METHODOLOGICAL DETAILS

By looking at (3)–(5) one can observe some significant similarities between two-point probability and two-point surface functions (see Fig. 1). The S_2 can be viewed as an autocorrelation of the image, i.e., computing correlations between shifted realization of the image [47]—something that can be used to effectively compute S_2 with the help of the FFT on modern hardware, especially GPUs. If we apply the same analogy to F_{SS} , Eq. (4) can be considered as an intersection of the interface with itself for all possible shifts [48]. Recall that for the correlation length of zero this intersection is technically infinity (Sec. II A). While surface-void function F_{sv} is well-defined at $r = 0$, it has some notable features. It can be calculated as cross-correlation between the interface and the void phase. Because cross-correlation is not commutative there is a complementary function, void-surface function F_{vs} , which is defined as cross-correlation between the void phase and the interface. Because these two functions are dependent, namely $F_{sv}(\mathbf{x}) = F_{vs}(-\mathbf{x})$, in this paper we will only speak of F_{sv} function.

A. The general algorithm

While we present and compare a bunch of slightly different methods to evaluate surface CFs, they are all based on a single general algorithm for each surface function as described below (A refers to the input 2D or 3D digital image, M is the

interface between two phases, and V is the image of the void phase only):

F_{SS} algorithm:

- 1: **procedure** SURFSURF (A, i)
- 2: $A' \leftarrow I^{(i)}(A)$ ▷ Apply indicator function to A .
- 3: $M \leftarrow M(A')$ ▷ Extract interface from A' .
- 4: **return** $\star(M, M)$ ▷ Autocorrelation of M .
- 5: **end Procedure**

F_{SV} algorithm:

- 1: **procedure** SURFVOID (A, i)
- 2: $A' \leftarrow I^{(i)}(A)$ ▷ Apply indicator function to A .
- 3: $M \leftarrow M(A')$ ▷ Extract interface from A' .
- 4: $V \leftarrow I^{(\text{void})}(A')$ ▷ Extract void phase from A' .
- 5: **return** $\star(M, V)$ ▷ Cross-correlation of M and V .
- 6: **end Procedure**

The algorithm is very general and allows utilizing different techniques for interface extraction or computation of the correlations, as detailed below. The cross-correlation function is introduced in Sec. III C. Consecutive stages of our method are shown in Fig. 3. The programming implementation details are provided in the Appendix B.

B. Interface extraction

We shall consider two methods for the interface extraction—the first is a naive approach, while the second is the one we propose to routinely apply for XCT and SEM images analysis.

1. Distance map

One solution to extract an interface between phases from an image is to use the distance map (also called the distance transform). Distance transform is a mapping from a binary image to a scalar field using one of the following functions:

$$D_o(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \in V_{\text{void}}, \\ \min \rho(\mathbf{x}, \mathbf{y}) \forall \mathbf{y} \in V_{\text{void}} & \text{otherwise,} \end{cases} \quad (8)$$

$$D_i(\mathbf{x}) = \begin{cases} 0 & \mathbf{x} \notin V_{\text{void}}, \\ \min \rho(\mathbf{x}, \mathbf{y}) \forall \mathbf{y} \notin V_{\text{void}} & \text{otherwise,} \end{cases} \quad (9)$$

where V_{void} is a set containing elements belonging to the void phase and $\rho(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between points \mathbf{x} and \mathbf{y} . Now $M(\mathbf{x})$ in Eq. (1) may be written as follows:

$$M(\mathbf{x}) = \begin{cases} 1 & 0 < D(\mathbf{x}) \leq \sqrt{d}, \\ 0 & \text{otherwise,} \end{cases}$$

where d is dimensionality of the image ($d = 2$ for 2D image, $d = 3$ for 3D image) and $D(\mathbf{x})$ is evaluated using either Eq. (8) or Eq. (9). When $D = D_i$ we call the resulting interface “the inner interface” and when $D = D_o$ we call it “the outer interface.” Hence, the inner interface consists of elements in the void phase which lie on the interface with the solid phase, and the outer interface is the other way around—it consists of elements in the solid phase which lie on the interface with the void phase.

2. Image filtering

This method creates a grayscale image of the interface with the help of the edge-detecting filter. The filter works by convolving the input image with signals Eq. (10) for the 2D case, or Eqs. (11)–(13) for the 3D case:

$$F = \frac{\sqrt{2}}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad (10)$$

$$F_{ij0} = \frac{\sqrt{2}}{27} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad (11)$$

$$F_{ij1} = \frac{\sqrt{2}}{27} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -26 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad (12)$$

$$F_{ij2} = F_{ij0}. \quad (13)$$

These signals are so called high-pass finite impulse response filters. Coefficients in tensors Eq. (10) and Eqs. (11)–(13) sum to zero, hence a result of application of an edge-detecting filter to a constant input is also zero; this also means there is no “interface” in a constant image. However, when an input image varies a lot, application of this filter gives a big response in place of variation, effectively detecting an edge. Scaling coefficients in front of the tensors are found empirically to achieve good correspondence with the theory.

An application of abovementioned filters (i.e., convolution) can be easily parallelized on GPU, and we do this by writing native Cuda code. After the convolution step is finished we take absolute values of the result as an approximation for $M(A')$ [recalling that $A' = I^{(\text{void})}(A)$ where A is the input image]. In physical sense the result of this procedure is proportional to the probability that a pixel/voxel under consideration belongs to the interface. From now on we shall call the filter F simply “edge-detecting filter” when no other filter is explicitly mentioned.

C. Computing correlations

Let us now describe the approach for the computation of cross-correlation function between two images f and g . In the case of autocorrelation function we assume $f = g$. Algorithm is straightforward when boundary conditions are periodic. For nonperiodic boundary conditions it is slightly more involved.

Periodic boundary condition

- 1: **procedure** $\star(f, g)$
- 2: $\hat{f} = F(f)$
- 3: $\hat{g} = F(g)$ ▷ Compute FFT of the input.
- 4: $\hat{cc} \leftarrow \hat{f} \cdot \hat{g}$ ▷ Multiply element-wise.
- 5: $cc \leftarrow F^{-1}(\hat{cc})$ ▷ Compute IFFT of \hat{cc} .
- 6: **return** cc divided by the number of pixels/voxels in the input.
- 7: **end Procedure**

Nonperiodic boundary condition

- 1: **procedure** $\star(f, g)$
- 2: Pad f and g with zeros in each dimension to the size $2n - 1$ where n is the size of the image in that dimension.
- 3: $\hat{f} = F(f)$
- 4: $\hat{g} = F(g)$ ▷ Compute FFT of the input.
- 5: $\hat{cc} \leftarrow \hat{f} \cdot \hat{g}$ ▷ Multiply element-wise.
- 6: $cc \leftarrow F^{-1}(\hat{cc})$ ▷ Compute IFFT of \hat{cc} .
- 7: circle shift cc by $n - 1$ in each direction, so index range becomes from $-(n - 1)$ to $n - 1$.
- 8: $cc_{ij} \leftarrow cc_{ij}/q_{ij}$ where $q_{ij} = (n - |i|)(n - |j|)$ ▷ Divide each element cc_{ij} by the number of pixels/voxels whose coordinate difference is equal to (i, j) .
- 9: **return** cc .
- 10: **end Procedure**

The results of the computations are a full correlation map. To obtain the ensemble averaged surface correlation functions one just needs to convert the map to CF versus r relationship. This is also possible to compute directional surface CFs in this manner in case the structure at hand is anisotropic. Without loss of generality, $\star(f, g)$ can be also computed by scanning with the line segment of length r , but this approach is more efficient on CPU as compared to GPU implementation here. If not stated otherwise, then we report the ensemble averaged functions computed from the full map (assuming the isotropy of the input image). One can easily compute directional CFs [12,13] using both segment scanning or from the correlation maps.

IV. APPLICATION TO SYNTHETIC IMAGES, VERIFICATION, AND BETTERMENT OF THE METHODOLOGY

A. Comparisons against analytical solutions

To evaluate the accuracy of surface CFs computations the most straightforward way is to compare them against analytical solutions. We use classical analytical solutions for overlapping disks and balls with fixed radius R and centers generated by Poisson point process with parameter λ . Starting with an image with resolution of 4096×4096 pixels, we then downscale it by 4, 16, and 64 times with the help of bicubic interpolation [3] (see Fig. 4). We show that calculated correla-

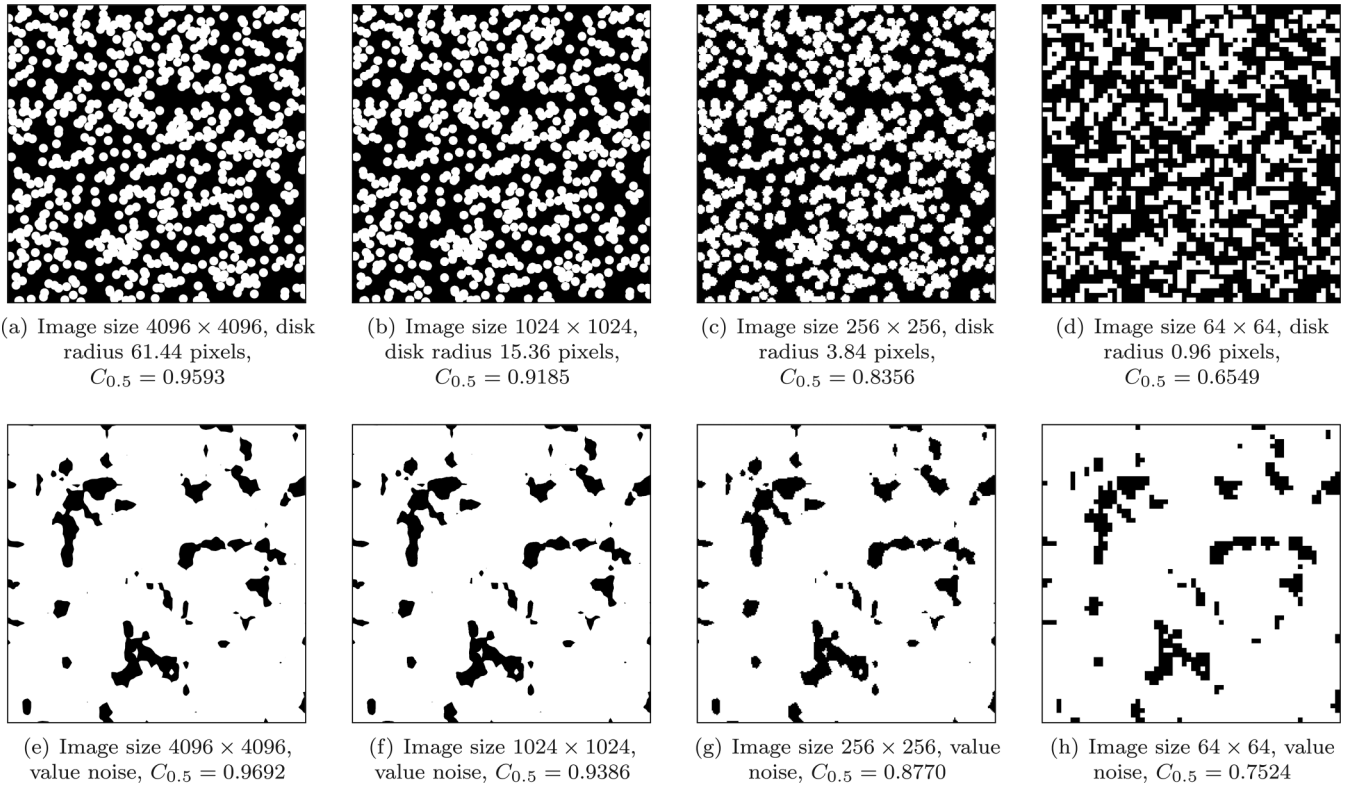


FIG. 4. 2D images of overlapping disks (a–d) and value noise (e–h) with different resolution (Sec. IV).

tion functions approach theoretical values when the resolution is high enough.

It is easy to show that for $a > 0$,

$$a^2 F_{SS}(a\mathbf{x}, a(\mathbf{x} + \mathbf{r})) = F_{SS}(\mathbf{x}, \mathbf{x} + \mathbf{r}), \quad (14)$$

$$a F_{SV}(a\mathbf{x}, a(\mathbf{x} + \mathbf{r})) = F_{SV}(\mathbf{x}, \mathbf{x} + \mathbf{r}). \quad (15)$$

Assuming that our images represent homogeneous and isotropic media, we calculate $F_{SS}(ar)$ and $F_{SV}(ar)$ for each original and rescaled image and multiply it by coefficient $a^2 = (L_{\text{scaled}}/L_{\text{orig}})^2$ and $a = L_{\text{scaled}}/L_{\text{orig}}$, respectively, where L_{scaled} is the side of the rescaled image and L_{orig} is the side of the original image. The resulting scaled surface CFs are shown in Fig. 5.

We observe two effects when downscaling the original image. The first effect is that $F_{SS}(ar)$ and $F_{SV}(ar)$ for down-scaled images lack in detail (i.e., they are less “noisy”). This can be easily explained by the interface between phases becoming “simpler” when resolution decreases. The second and much more important effect is that both correlation functions become largely underestimated with the decline in resolution. This latter effect is in general connected to the fact that the interface in digital images has nonnegligible thickness—i.e., the difference between real “continuous” interface versus “digital” interface (Fig. 1). It is logical to assume that with increasing spatial resolution the influence of this thickness will diminish. This is exactly what we observe on Fig. 5 where increasing disk discretization leads to convergence with analytical solution; the accuracy of the computed CFs is almost perfect for discretization of about 62 pixels for each disk’s diameter. A natural question arises:

“Is there a criterion of image quality (resolution) that allows to predict the quality of surface CFs evaluation from digital images?” Turns out there is a possibility to establish such an empirical criterion based on spectral analysis of input images and Sec. IV D will provide all necessary details.

B. Convergence of correlation functions with increase of resolution

Now consider a situation when an image is obtained by taking samples of a function $f : \mathbb{R}^n \rightarrow \{0, 1\}$. The samples are taken from a regularly spaced lattice which covers the range $[0, L]^n$ with interval Δ between samples. The resulting image has then $L/\Delta + 1$ pixels in each dimension. We will show empirically that for any sequence Δ_k so that $\Delta_{k+1} < \Delta_k$ and $\Delta_k \rightarrow 0$ sequences $F_{SS}^{\Delta_k}(r)$ and $F_{SV}^{\Delta_k}(r)$ computed for images with lattice interval Δ_k converge to some limit functions $F_{SS}(r)$ and $F_{SV}(r)$.

An example of f is a thresholded value noise function. A value noise is a procedurally generated noise which in one-dimensional case works as follows. First, define a family of regularly spaced lattices U_k with points in coordinates $2^{-k}n$, $n \in \mathbb{Z}$. Each point of a lattice is assigned a random value in the range $[0, 1]$. Then for each U_k we define a function $g_k(x) : \mathbb{R} \rightarrow \mathbb{R}$ which interpolates random values in U_k linearly. This value noise function is defined as follows:

$$g(x; n) = \frac{\sum_{k=0}^{n-1} 2^{-k} g_k(x)}{\sum_{k=0}^{n-1} 2^{-k}} = \frac{2^n}{2^{n+1} - 2} \sum_{k=0}^{n-1} 2^{-k} g_k(x).$$

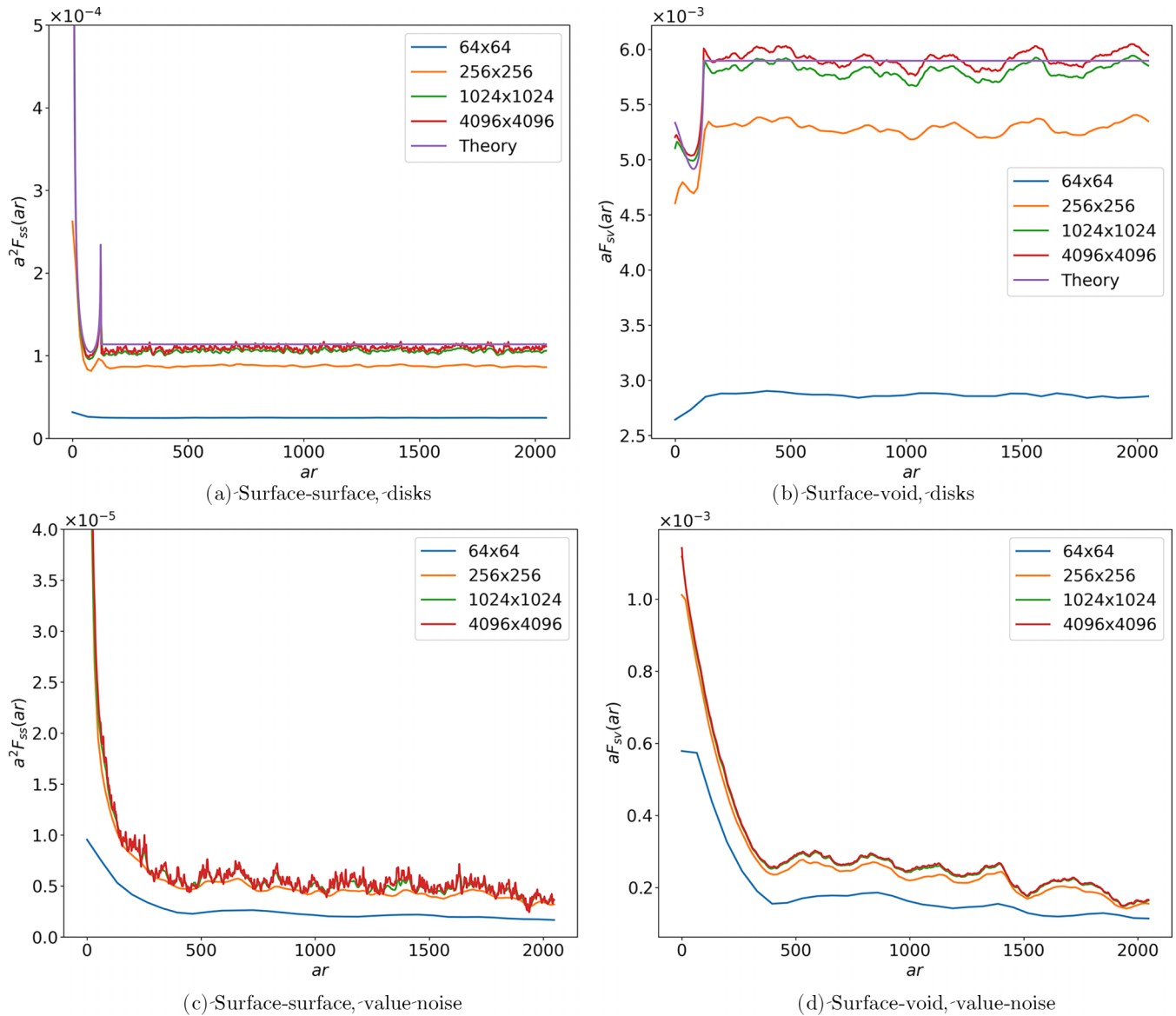


FIG. 5. Surface-surface and surface-void correlation functions for images of overlapping disks and thresholded value noise obtained with different resolution. Theoretical results are also shown for overlapping disks. Image filtering was used for interface extraction (10).

Here n is a parameter called the number of octaves. A thresholded noise is then

$$f(x; n) = \Theta(g(x; n) - 1/2),$$

where $\Theta(x)$ is the Heaviside step function. A generalization for higher dimensionality is constructed using the same principle. An example of thresholded value noise with three octaves is depicted on Fig. 4. Correlation functions calculated for the noise with $n = 3$ at different resolutions are on Fig. 5. If we calculate surface-surface and surface-void correlation functions for these images and scale them as described above, we again converge to some “true” correlation functions for continuous function f .

C. Effects of image magnification

As we have just observed for analytical Poisson disks and arbitrary structures based on thresholded value noise, with

increasing resolution (or, in other words, decreasing coarseness) of the image, surface correlation functions converge to “infinite resolution” values that can be seen as a target of CFs computations. However, if the image at hand is of limited resolution, then it would be very practical to know if rescaling this image with increased resolution can lead to acceptable surface-surface functions evaluation. To explore such a possibility we tried different methods to magnify the image: bilinear, bicubic, and nearest-neighbor interpolation.

To have true CFs for a comparison, we again utilize analytical Poisson disks: a low-resolution and high-resolution images were produced by discretization of disks at known positions, and then the low-resolution image was magnified. Because the resized image becomes grayscale, we use the following function as an indicator for the void phase provided the void phase is marked as 0 in the input image:

$$I^{(\text{void})}(x) = (1 - x)^{1.5}.$$

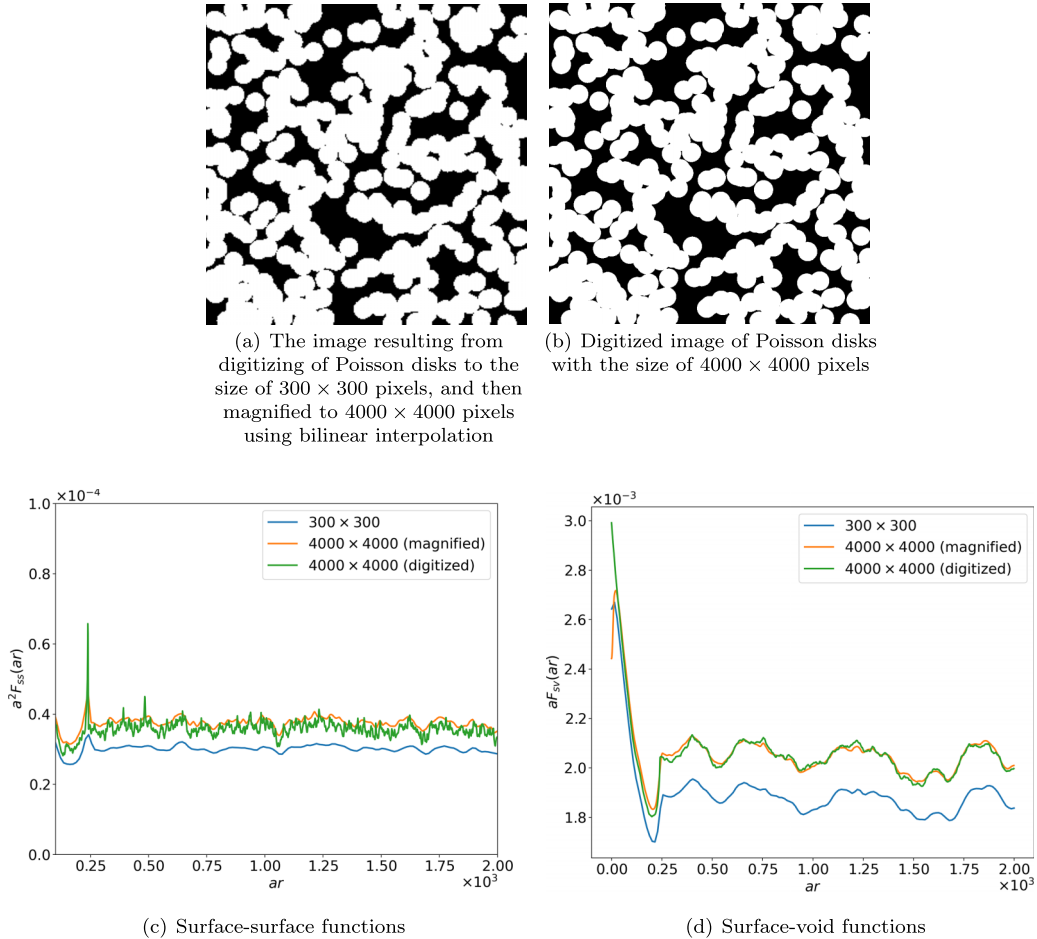


FIG. 6. An example of images with almost identical correlation functions obtained using different methods: direct digitalization from known disk positions and magnification using bilinear interpolation. Image filtering was used for interface extraction (10).

We found that all interpolation methods are good for resizing purposes with exception of the nearest-neighbor approach. As an example, Fig. 6 shows the images of overlapping disks with almost the same surface-surface and surface-void functions. The values of CFs are almost identical for high-resolution and magnified images; the only flaw being the difference in F_{SS} for resized and digitized images due to pixelization of the interface (the absence of “noise” on F_{SS} for the rescaled image).

D. Criterion for accurate evaluation of surface CFs from discrete images

Let us define one-dimensional forward (F) and inverse (F^{-1}) Fourier transforms as

$$\hat{f}(z) = F[f](z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-i\pi xz} dx, \quad (16)$$

$$f(x) = F^{-1}[\hat{f}](z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(z) e^{i\pi xz} dz. \quad (17)$$

Here $f(x)$ and $\hat{f}(z)$ can be thought of as representations of a signal in the time domain and the frequency domain, respectively. Equations (16) and (17) preserve norm on L_2 : $(f, f) = (\hat{f}, \hat{f})$. It is said that Fourier transform preserves “energy” of the signal. According to Riemann-Lebesgue lemma [49]

energy of any signal “concentrates” around low frequencies. A measure of how much energy concentrated in low frequencies (for some definition of low frequencies) is a key to the understanding of the problem of correctness of our approach. The Shannon sampling theorem [49] states that it is possible to reconstruct a band-limited signal $f(x)$ whose frequencies lie in the range $[0, f]$ from a sequence of samples $\{f_n\}$ if the sampling rate is no less than $2f$. We introduce a parameter C_a :

$$f_0(x) = f(x) - \langle f(x) \rangle,$$

$$C_a = \frac{\int_{-a\omega}^{a\omega} |\hat{f}_0(z)|^2 dz}{\int_{-\omega}^{\omega} |\hat{f}_0(z)|^2 dz},$$

where $\omega = 2\pi f$ and $\langle f(x) \rangle$ is the mean value of $f(x)$ over its domain. The criterion for correctness is then

$$C_a > 1 - \xi$$

for some a and ξ . This criterion tells us exactly how much energy in $f_0(x)$ is concentrated in a low frequency range $[0, af]$ compared to the whole range $[0, f]$. Here we propose $a = 0.5$ and $\xi = 0.07$ as a strict criterion which is based on results of Sec. IV A. For overlapping disks we immediately observe that the image with resolution 4096×4096 has surface CFs very close to analytical solution [Figs. 5(a) and 5(b)] and perfectly satisfy our $C_{0.5}$ criterion. The other three downsampled

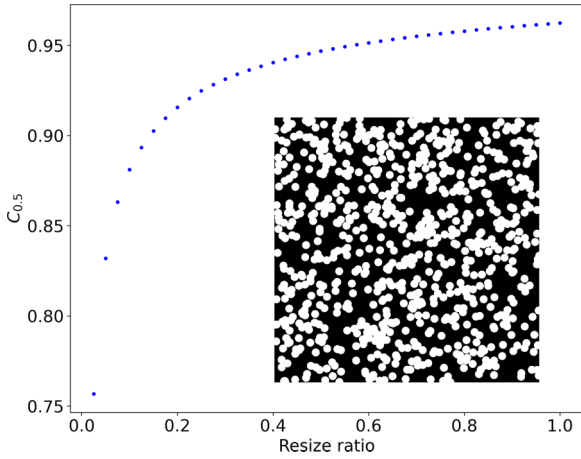


FIG. 7. Decay of criterion $C_{0.5}$ as a function of the image resolution for the downscaled image (shown as a resize ratio). The image used for the study is shown as an inset.

images fail to do so. The first two images of value noise (with resolution 4096×4096 and 1024×1024) pass the criterion and the other two do not, which is in agreement with Figs. 5(c) and 5(d) (correlation functions converge at resolution about 1024×1024).

Thus, we conclude after inspecting Fig. 4 and Fig. 5 that the criterion correctly selects images which are suitable for calculation of surface correlation functions. The criterion $C_{0.5}$ decays fast with a decline in the image resolution as demonstrated in Fig. 7.

E. The choice of method to extract the interface

Finally, we explore all major edge-detecting filters, including “naive” distance map approach. Surface-surface functions for overlapping Poisson disks ($R = 10$, $\lambda = 2 \times 10^{-3}$ on image with dimensions of 1000×1000 pixels) computed with different methods of interface extraction are shown on Fig. 8. These methods include image filtering with filter Eqs. (10)–(13) and interface extraction with distance map. When using distance map both outer [Eq. 8] and inner [Eq. 9] interfaces are used to compute the correlation function. An average over the last two functions is also shown. Edge-detecting filter clearly outperforms the distance map approach. However, the result on Fig. 8 could be related to the resolution of studied image. If we now increase the discretization of the disks and, thus, the image resolution ($R = 70$, $\lambda = 3 \times 10^{-6}$ on image with dimensions of 5000×5000 pixels), then we get the result as presented on Fig. 9. In this case the image filtering method again collapses to an analytical solution whilst the distance map method failed to do so. This subsection’s results clearly advocate in favor of the edge-detecting filter in case of surface CFs evolution based on digital images.

To demonstrate the possibilities to compute directional surface correlation functions, on Fig. 10 we show a comparison of the ensemble F_{SS} computed by averaging the full correlation map against the CF computed along a single direction (the input image is isotropic Poisson disks from Fig. 8). As expected, less sampling results in more noise but in gen-

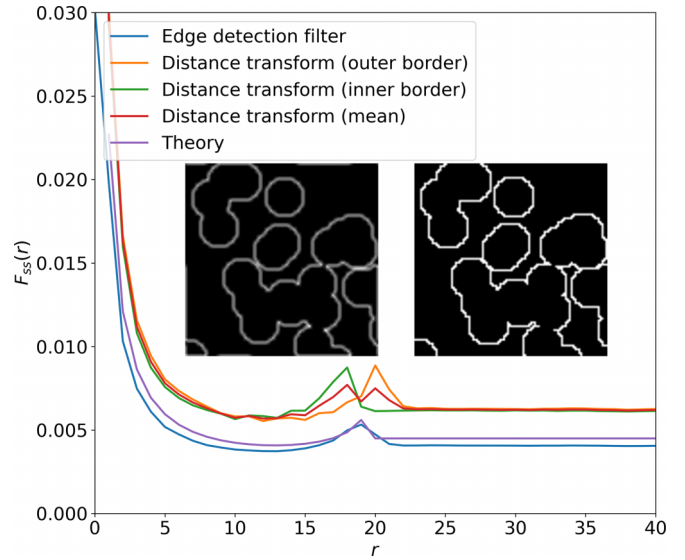


FIG. 8. Surface-surface correlation function calculated with different interface extraction methods (Edge-detection filter to the left, and distance map to the right on the insets). Discretization of each disk is 20 pixels per diameter, the size is 1000×1000 pixels.

eral coincide with analytical and ensemble average computed CFs.

In addition to edge-detecting filter of our choice, there are plenty of other kernels for gradient evaluation [50–52]. In total, we investigated Sobel, Ando, Sharr, Bickley, and Prewitt filters. Their performance is almost the same (Fig. 11). The errors against the analytical solution are very similar and the overall performance depends on the particular image. Therefore, it is difficult to recommend specific kernel; so, the choice of filter Eqs. (10)–(12) is purely due to the fact that it is computationally more efficient than gradient searching filters.

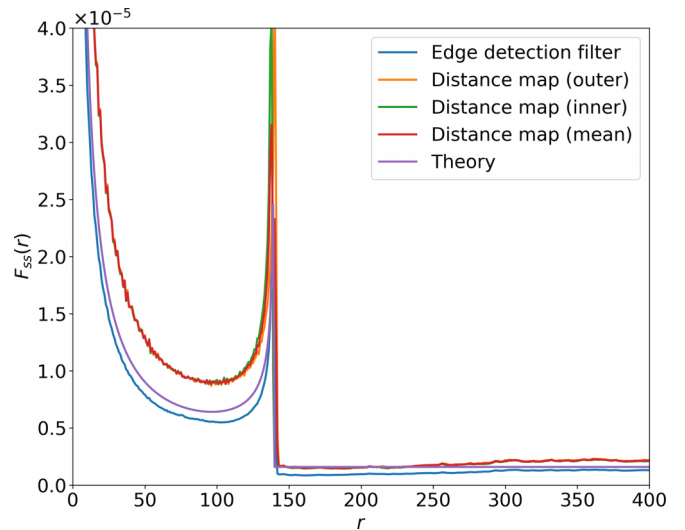


FIG. 9. Comparison of edge-detecting filter and distance map approach on high-resolution image with 140 pixels per disk diameter, the size is 5000×5000 pixels.

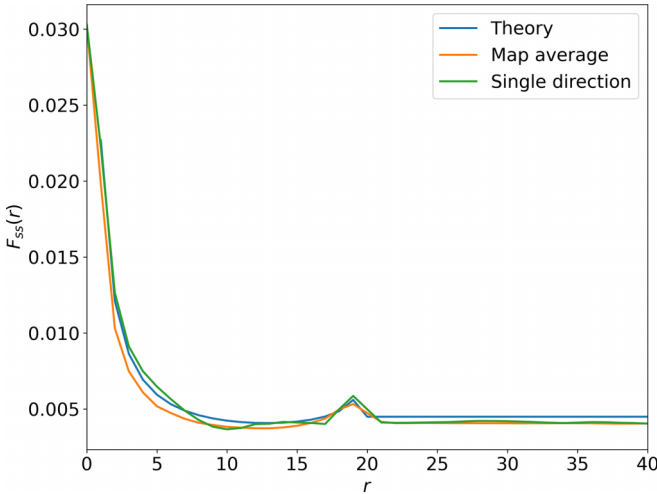


FIG. 10. The comparison of directional and ensemble averaged over the full map F_{SS} .

V. COMPARISON AGAINST AN EARLIER ALGORITHM OF MA AND TORQUATO

To establish a novel methodology, one needs to prove that it is somehow more advantageous against existing approaches. In this section, we compare our method with earlier algorithm of Ma and Torquato [38]. The method as proposed in this paper has some distinct advantages:

(1) It has better complexity than the earlier approach. There are two steps in our algorithm: edge detection and the autocorrelation computation. The first step has $O(n)$ complexity and the second step has $O(n \log n)$ complexity where n is a number of pixels (or voxels) in the input image. Ma and Torquato’s algorithm has $O(n)$ complexity for finding intersections with probe rays and $O(m^2)$ for computation of correlation functions where m is a number of intersections found. Therefore, performance of their algorithm depends on

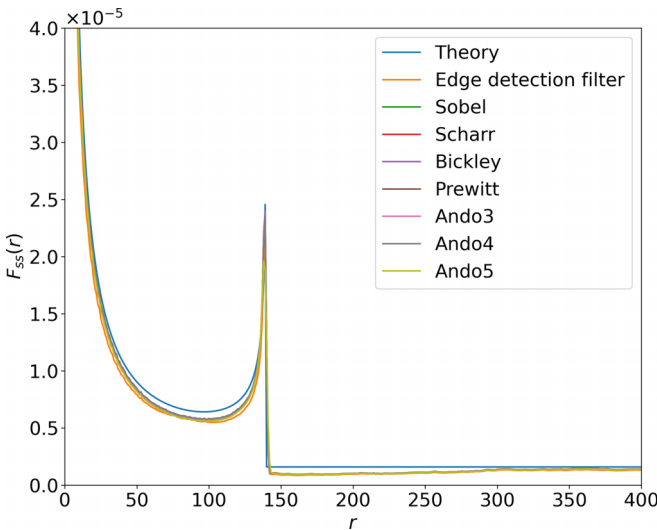


FIG. 11. Comparison of different edge-detecting filters in their ability to highlight the interface for surface-surface CFs computation (only the case of F_{SS} is shown due to similarity of the results for F_{SV}).

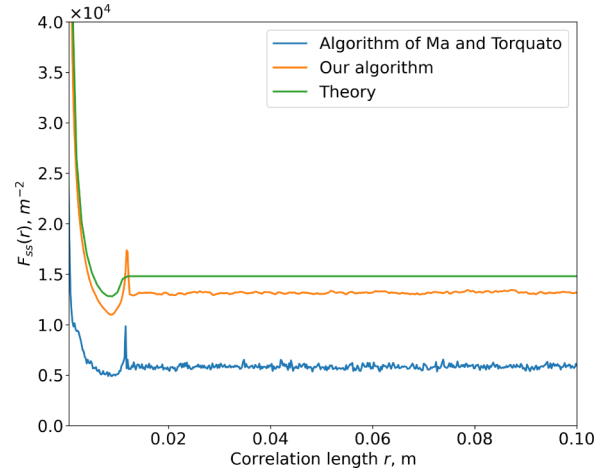


FIG. 12. Our algorithm compared against Ma-Torquato algorithm. The input image is the one of 10 000 overlapping disks, image dimensions are 5000×5000 , ratio of solid phase $\phi = 0.675$, $C_{0.5} = 0.934$. The value of $b = 0.025D$ was used in Ma-Torquato algorithm—exactly what was used in their work to obtain the most accurate results.

the input (more specifically, on the number of intersections of the interface and probe rays) and may be unsatisfying when $n \sim m$.

(2) Our approach produces less noise on computed surface function curves. In case of Ma-Torquato method, the noise persists even on high resolution image (Fig. 12).

(3) In the case of overlapping disks, our algorithm produces results which are closer to the theory than earlier method (Fig. 12). Note that here we refer to “digital” approach only (as the algorithm by Ma and Torquato can be used to obtain the exact continuum representation of the overlapping disk problem).

(4) Our algorithm has less inner parameters that are hard to justify. Ma-Torquato algorithm is very sensitive to selection of standard deviation b for Gauss filter (Fig. 13), unless tuned to known specific surface and porosity values. Moreover, it implements single thresholding segmentation (see more discussion below).

(5) Due to the chosen filter and autocorrelation computation algorithms our method is easily parallelizable on both CPU and GPU architectures.

In their original paper, Ma and Torquato claim that their algorithm can be applied to grayscale images directly. In authors’ implementation a single thresholding segmentation is applied with a parameter T tuned to known specific surface and porosity, and is defined as follows:

$$I^{(\text{solid})}(\mathbf{x}) = \Theta(A(\mathbf{x}) - T),$$

where $A(\mathbf{x})$ is an intensity of a pixel in the image at the coordinate \mathbf{x} . Tuning T gives accurate results in case specific surface and porosity are known a priori, but is highly sensitive to a choice of T (Fig. 14). For experimental images these parameters are not known, and even if obtained experimentally (for example, using porosimetry methods) would lead to wrong computations due to numerous imaging resolution effects discussed in Sec. I. This highlights the necessity of advanced

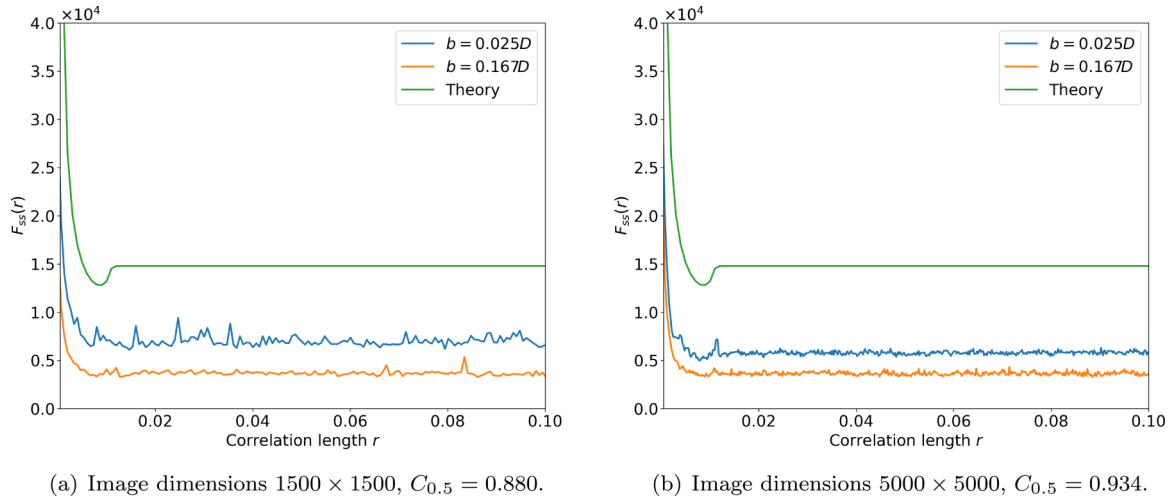


FIG. 13. Illustration of how a choice of b affects the results of Ma-Torquato algorithm. Two cases of images of overlapping disks are considered.

XCT or SEM images segmentation techniques [42,53]—it is a well-established fact in porous media research area that single thresholding techniques produce vastly inaccurate segmentations for such experimental images [54,55].

Implementation of Ma-Torquato algorithm in Julia language is available [56] for future applications and benchmarks.

VI. APPLICATION TO BINARY XCT AND SEM IMAGES OF POROUS MEDIA

After the verification of our computational approach based on analytical solution and establishing the criterion $C_{0.5}$ we now have enough tools to evaluate surface CFs for different porous media images, including real XCT and SEM data.

A. Porous media images collection

To test our surface CFs computational framework we collected a variety of 3D XCT and 2D SEM images of two types

of natural porous media: sandstones and carbonate rocks. The former are expected to have most of their porosity visible on XCT scans, except for nano-porosity associated with the clay that can clog the pore space. Carbonate rocks are monomineral samples, but usually exhibiting hierarchical structure, i.e., XCT scanning can reveal only pore sizes with the range within the imaging resolution, whilst imaging with higher resolution (e.g., SEM or FIB-SEM techniques) usually reveals submicron (under-resolution porosity for XCT) porosity. The relevant details on the samples are provided in table I.

B. Surface correlation functions for the studied collection

In Figs. 15 and 16 we report computed F_{SS} and F_{SV} correlation functions for a variety of XCT and SEM images, respectively. Almost all these cases satisfy our $C_{0.5}$ criterion and, thus, provide a possibility to interpret resulting CFs.

Images on Fig. 17 fail to satisfy our $C_{0.5}$ criterion. It is possible to apply the magnification technique to improve

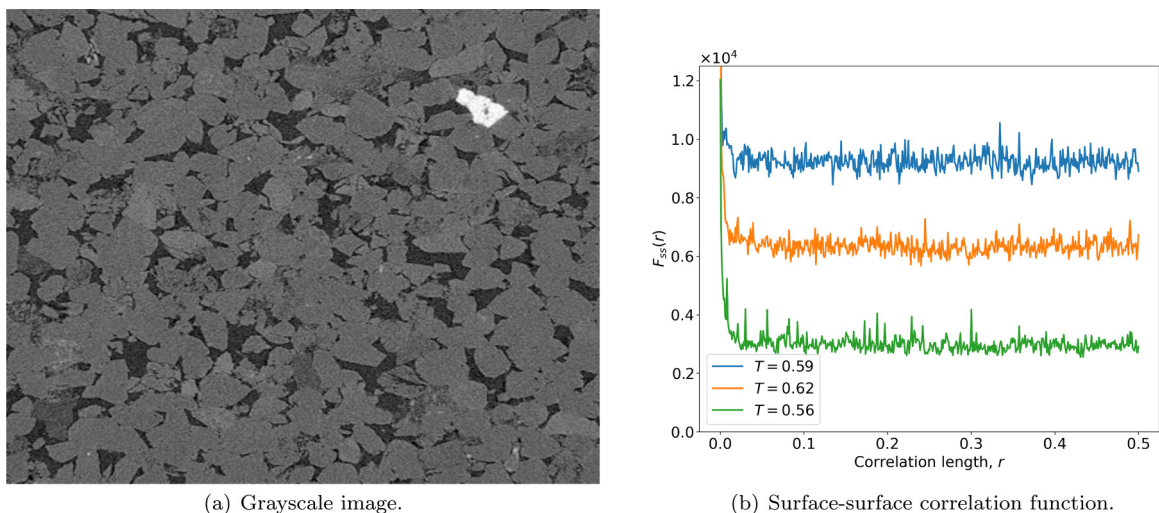


FIG. 14. A two-dimensional slice of coarse-grained sandstone (a) and surface-surface correlation functions computed with different threshold parameter T (b).

TABLE I. Summary of samples used for calculation of surface correlation functions.

Sample name	Image type	Dimensions (pixels)	Image resolution (μm)
Sandstone 1	XCT	$500 \times 500 \times 500$	3.02
Sandstone 2	XCT	$500 \times 500 \times 500$	3.02
Sandstone 3	XCT	$500 \times 500 \times 500$	3.02
Carbonate 1	XCT	$500 \times 500 \times 500$	3.02
Carbonate 2	XCT	$500 \times 500 \times 500$	3.02
Carbonate 3	XCT	$500 \times 500 \times 500$	3.02
Sandstone 4	SEM	1280×869	0.10
Sandstone 5	SEM	1280×869	0.05
Carbonate 4	SEM	1024×691	0.06
Carbonate 5	SEM	1024×691	0.29
Carbonate B1	SEM	1024×691	0.57
Carbonate B2	SEM	1024×691	1.09
Carbonate B3	XCT	$500 \times 500 \times 500$	1.07
Carbonate B4	XCT	$500 \times 500 \times 500$	1.07

$C_{0.5}$, but this lacks physical meaning—from our experience we can immediately tell that these rocks possess significant under-resolution porosity that would reveal itself on higher-resolution images. We discuss the strategies for correct evaluation of surface correlation functions for such porous media in Sec. VII.

In the general case, the correlation functions of the boundary of a binary set have at least two characteristic properties: the limit of F_{ss} at large distances is related to the square of the surface area (in voxels), and the rate of decrease of the correlation function near zero characterizes the “memory” of the random process describing the boundary. The last property can be described by introducing the concept of correlation radius—the characteristic attenuation distance of the correlation function. Figure 15 shows examples of XCT images of sandstones and carbonates along with surface correlation functions calculated using developed methodology. One can see from the graphs presented that the right limit for all sandstones is significantly higher than for carbonate materials, which corresponds to a larger surface area for these cubes (in voxels). However, the correlation boundary radius for carbonate samples is larger than for sandstones, which indicates a greater surface smoothness.

We consider it important to mention one more surface parameter, which can be easily calculated based on the presented methodology. One of the key topological properties of rough surfaces for physics and applications is the fractal dimension. Let us divide the micromodel into cubes with edge length ε (in voxel). The topological measure of the surface is defined as

$$M_d(\varepsilon) = N(\varepsilon)\varepsilon^d,$$

where $N(\varepsilon)$ is the number of cubes covering at least one point of the surface. The fractal dimension is defined as the value of d for which $M_d(\varepsilon) \rightarrow \text{const} \neq 0$ for $\varepsilon \rightarrow 0$. In the case of piecewise smooth line or surface, the fractal dimension is an integer equal to 1 or 2, respectively (coincides with the topological dimension). For an arbitrary surfaces, the fractal dimension is not an integer (see, for example, Ref. [57], where fractal dimensions of real rock pore surfaces are calculated and analysed). The fractal dimension can be

effectively calculated using the box covering algorithm. For each ε , the number $N(\varepsilon)$ of cubes covering the pore surface is calculated; fractal dimension d is estimated from the slope of the curve describing $\ln N(\varepsilon)$ as a function of $\ln \frac{1}{\varepsilon}$. Thus, calculating the correlation functions F_{ss} of the surface for different spatial sampling scales ε (see Fig. 5 with our results of surface-surface and surface-void correlation functions for images obtained with different resolution) and analyzing the asymptotic at large distances gives us the value of $N^2(\varepsilon)$. This makes the further calculation of the fractal dimension an easy technical procedure.

It has been shown in many works (see, for example, Refs. [58–60]) that the fractal dimension of a surface as a measure of roughness can make a significant contribution to such physical phenomena as the thermodynamics of adsorption and materials storage properties, the formation of thin films, surface diffusion, and adhesion. However, further development of this issue is not the subject of this article and requires a separate study, so we only briefly discuss it in this article as a further possibility.

C. Computational efficiency

We measured the execution times of functions from our `CorrelationFunctions.jl` package to compute F_{SS} and F_{SV} . The hardware we used is Intel Xeon Gold 6248R CPU with DDR4 2400 MHz RAM and Nvidia GeForce RTX 3090 GPU running under Linux. Input data consists of 2D and 3D square bit arrays with side varying from 1000 to 10000 with step 1000 (2D case) or from 100 to 1000 with step 100 (3D case). The obtained wall times are presented on Fig. 18. As our video RAM is limited, the GPU can process three dimensional arrays with dimensions no more than $700 \times 700 \times 700$ voxels. A observed speedup from the use of GPU is about 30 times as compared to CPU computations.

VII. DISCUSSION AND OUTLINE

The “digital” approach for applications to real XCT and SEM images as developed in this paper has some important limitations. Namely, the magnification-based correction of

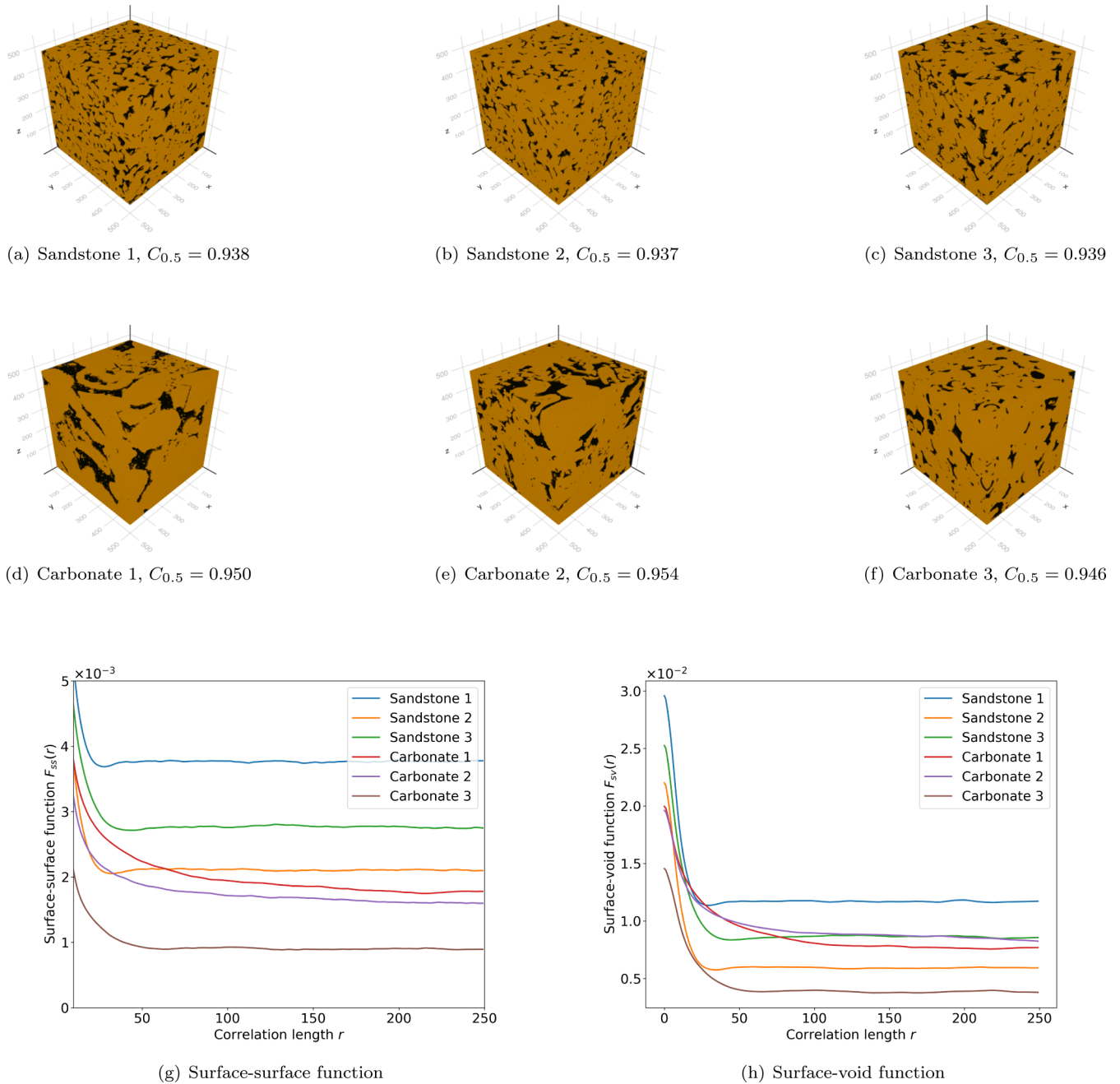


FIG. 15. Examples of XCT images along with surface correlation functions calculated using developed methodology.

coarse images (Sec. **IV C**) or the $C_{0.5}$ criterion (Sec. **IV D**) provide accurate surface correlation functions only in case the under-resolution porosity is absent in the image. This means that accurate CFs can be sampled only for the case that with increasing resolution no new details appear on the image. The ideal imaging resolution is not known a priori and, thus, cannot be assessed with $C_{0.5}$. For sandstones, we expect accurate surface CFs evaluation, but for carbonates on Fig. 17 the presence of unresolved porosity is expected. For other carbonates such as Carbonates 1–3 the presence of such porosity is likely, and for Carbonates 4–5 the resolution is probably optimal—however, both of these hypotheses needs to be tested by imaging with a higher resolution. Another way would be to compare “digital” CFs with those obtained

experimentally using small angle scattering (note that current generation synchrotrons are operating in the limit $\lesssim 1 \mu\text{m}$ resolution). However, the surface CFs we obtain are correct under the assumption of no subresolution details. This is an important achievement, as the problem of under-resolution porosity can be then solved by multiscale image analysis [15,61].

Considering the accuracy of the magnification procedure, it is also closely connected to the under-resolution porosity problem. On Fig. 6 it was clear that magnification can result in correct values of surface CFs, but for F_{SS} some important (subresolution) information is simply not available. We argue that this unfortunate downside can be potentially mitigated by higher-resolution imaging of the interface in question with subsequent correction of the magnified surface-surface func-

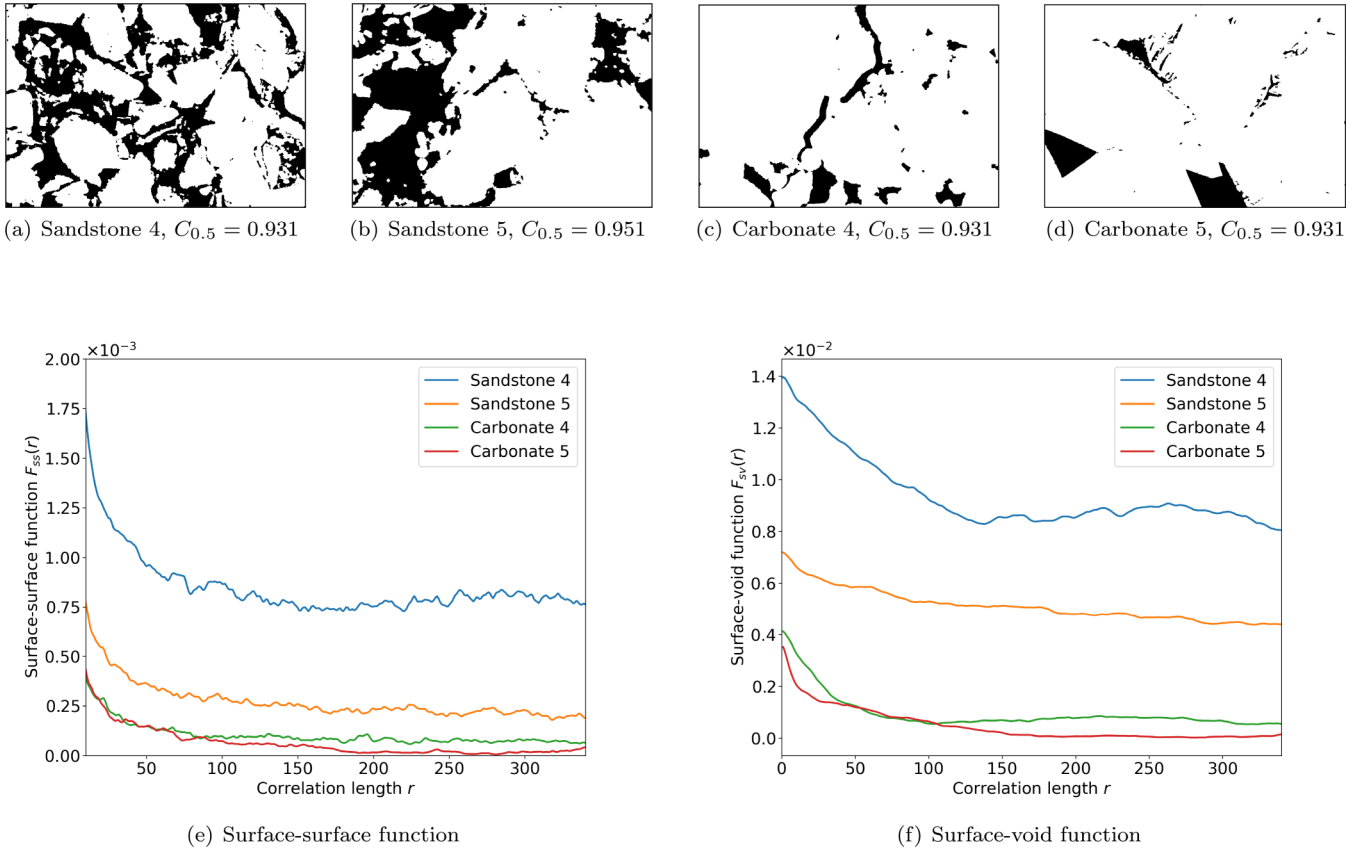


FIG. 16. Examples of surface correlation functions evaluated for SEM images of real porous media.

tion. Another interesting approach would be to relate the F_{SS} to some roughness coefficient that could change with time due to, for example, reactive transport within the pores [62–64] or due to mechanical load. Thus, in conjunction with high-resolution imaging F_{SS} and F_{SV} could serve as excellent descriptors of surface change dynamics [65].

We described two ways to extract the interface to compute surface CFs—“naive” approach based on distance map transform, and using edge-detecting filter. While we showed that the latter approach is much more accurate (e.g., Fig. 9), the naive one is still useful. In case we want to utilize F_{SS} and F_{SV} for stochastic reconstructions [23,26,30] we advocate the usage of distance map transform; moreover, the choice of inner or outer interface (Fig. 8) is not important. The reason is the procedure of CFs update during the simulated annealing process. The optimization of the filter-based approach is more involved and requires updates of the filter within some area first, and then local CFs recalculations. However, the naive approach allows to utilize the classical straightforward and extremely efficient technique [66]. Considering that the exact values of the CFs are not important, the usage of distance maps introduces no inconsistencies, but improves the results of the reconstruction.

In addition to fundamental XCT and SEM images acquisition problems (such as inhomogeneous chemical composition and density of the solid phase, and partial volume effects), as discussed in Sec. I, the “digital” approach to evaluate surface correlation functions enjoys some additional advantages. As Sec. IV C demonstrated, digital computations can be fast—

this can be achieved by FFT on modern GPU hardware, or by vectorized linear segment scanning [13]. In addition to efficient computations, digital methods allow efficient computations of higher order statistics, such as n-point correlation functions [67]. Such efficiency and flexibility in computations is not available for “continuous” approach. This statement is by no means a critique of such an approach, as it allows to compute the exact surface CFs in cases when the structure can be described by some continuous function, e.g., Gaussian random field [38]. Aforementioned computational efficiency opens numerous ways to apply surface correlation functions in description of real surfaces, surface structural dynamics and establishes them as universal interface descriptors. As such, F_{SS} and F_{SV} can serve as metrics in deep learning application for stochastic reconstructions and especially super-resolution, e.g., Refs. [68–70], or can be, likely, even incorporated into the cost function during training on modern GPUs.

VIII. SUMMARY

In this contribution we proposed a computationally efficient and very flexible “digital” framework to evaluate surface correlation functions for XCT and SEM images of real porous media and materials. The methodology not only allows to compute the full correlation maps in addition to directional or ensemble average CFs, but also does this on both CPU and GPU architectures. The code to perform all computations is fully available as a Jupiter notebook (see Supplemental Material [71]); moreover, the full functionality is now a part

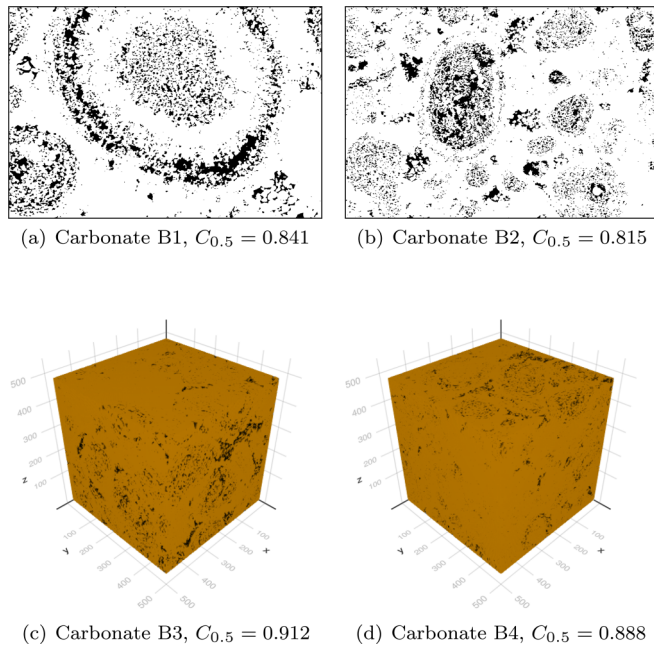


FIG. 17. Examples of SEM and XCT images which do not satisfy our $C_{0.5}$ criterion.

of the freely available `CorrelationFunctions.jl` package. Based on the computed F_{SS} and F_{SV} functions for a variety of artificial and real XCT/SEM digital images we were able to conclude:

- (1) On digital images edge-detecting filters can be utilized to effectively extract the interface for surface CFs computations.
- (2) It is possible to use digital images as a foundation for robust surface correlation functions evaluation in case their spatial resolution is adequate.
- (3) It is possible to establish $C_{0.5}$ criterion to evaluate if the quality of the image at hand is adequate; for artificial images passing this criterion the analytically known and computed CFs are similar.

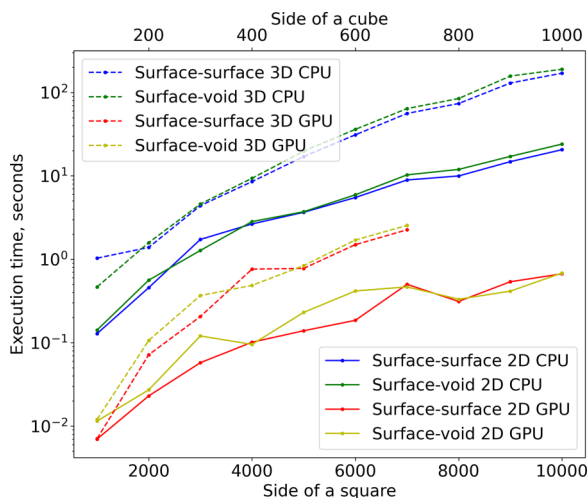


FIG. 18. Execution times for 2D and 3D input images for CPU and GPU implementations.

(4) In case the $C_{0.5}$ criterion is below the desired value, one can utilize image magnification (e.g., in the form of bicubic interpolation) to increase the image resolution and achieve the necessary $C_{0.5}$ criterion—in this latter case the surface CFs computed from the magnified image reach high accuracy.

The developed methodology allows applying surface CFs to describe the structure of porous materials based on their experimental images and enhance stochastic reconstructions or super-resolution procedures, or serve as an efficient metrics in machine learning applications.

ACKNOWLEDGMENTS

This research was supported by Russian Science Foundation Grant No. 19-72-10082. Collaborative effort of the authors within the FaT iMP (Flow and Transport in Media with Pores) research group [72] and we used some of its software. We thank Prof. Salvatore Torquato for directing us to a derivation of analytical formulas for 2D Poisson disks.

APPENDIX A: DERIVATION OF ANALYTICAL SOLUTIONS FOR 2D POISSON DISKS

Analytic representation of surface-surface and surface-void correlation functions for overlapping n -dimensional balls with centers generated by Poisson point process is well known [47]:

$$F_{SV}(r) = - \lim_{a_1 \rightarrow R} \frac{\partial}{\partial a_1} e^{-\lambda S_{\text{tot}}(r, a_1, R)}, \quad (\text{A1})$$

$$F_{SS}(r) = \lim_{a_1, a_2 \rightarrow R} \frac{\partial}{\partial a_1} \frac{\partial}{\partial a_2} e^{-\lambda S_{\text{tot}}(r, a_1, a_2)}. \quad (\text{A2})$$

Here $S_{\text{tot}}(r, a_1, a_2)$ refers to a union volume of two n -dimensional balls of radii a_1 and a_2 with a distance r between their centers and λ is a parameter of Poisson process. For two-dimensional disks we have the following expression for $S_{\text{tot}}(r, a_1, a_2)$:

$$S_{\text{tot}}(r, a_1, a_2) = \pi a_1^2 + \pi a_2^2 - S_{\text{int}}(r, a_1, a_2), \quad (\text{A3})$$

where $S_{\text{int}}(r, a_1, a_2)$ is an intersection area of two disks, being equal to

$$\begin{aligned} S_{\text{int}}(r, a_1, a_2) &= a_1^2 \arccos\left(\frac{r^2 + a_1^2 - a_2^2}{2a_1 r}\right) \\ &+ a_2^2 \arccos\left(\frac{r^2 + a_2^2 - a_1^2}{2a_2 r}\right) - \frac{\sqrt{Y}}{2}, \\ Y &= (-r + a_1 + a_2)(r + a_2 - a_1) \\ &\times (r + a_1 - a_2)(r + a_1 + a_2), \end{aligned}$$

when $r < 2R$ and zero otherwise.

Substituting $S_{\text{int}}(r, a_1, a_2)$ into Eq. (A3) and applying Eq. (A2) for $n = 2$ we obtain Eq. (7) for $F_{SS}(r)$. Similarly applying Eq. (A1) for $n = 2$ we obtain Eq. (6) for $F_{SV}(r)$.

APPENDIX B: IMPLEMENTATION OF SURFACE FUNCTION COMPUTATIONS

The code used to compute surface correlation functions in this manuscript was written in Julia language and available as a Jupyter notebook in the Supplemental Material

[71]. It uses `CorrelationFunctions.jl` package developed by our group (paper describing all aspects of different CFs computation in addition to surface functions is currently in preparation) that allows efficient computation of

surface and other correlation functions using both CPU and GPU architectures. The package allows to compute all classical CFs described in Torquato's book [47] from digital images.

-
- [1] A. Cecen, Y. C. Yabansu, and S. R. Kalidindi, *Acta Mater.* **158**, 53 (2018).
- [2] P.-E. Chen, W. Xu, N. Chawla, Y. Ren, and Y. Jiao, *Acta Mater.* **179**, 317 (2019).
- [3] R. Ledesma-Alonso, R. Barbosa, and J. Ortegón, *Phys. Rev. E* **97**, 023304 (2018).
- [4] K. Gerke, M. Karsanina, and E. Skvortsova, *Eurasian Soil Sci.* **45**, 861 (2012).
- [5] M. V. Karsanina, K. M. Gerke, E. B. Skvortsova, and D. Mallants, *PLoS ONE* **10**, e0126515 (2015).
- [6] M. V. Karsanina, E. V. Lavrukhin, D. S. Fomin, A. V. Yudina, K. N. Abrosimov, and K. M. Gerke, *Eur. J. Soil Sci.* **72**, 561 (2021).
- [7] M. Takada and B. Jain, *Mon. Not. R. Astron. Soc.* **340**, 580 (2003).
- [8] A. Derossi, K. M. Gerke, M. V. Karsanina, B. Nicolai, P. Verboven, and C. Severini, *J. Food Eng.* **241**, 116 (2019).
- [9] K. M. Gerke, M. V. Karsanina, and R. Katsman, *Phys. Rev. E* **100**, 053312 (2019).
- [10] P. Čapek, V. Hejtmánek, J. Kolafa, and L. Brabec, *Transp. Porous Media* **88**, 87 (2011).
- [11] J. F. Thovert and P. M. Adler, *Phys. Rev. E* **83**, 056116 (2011).
- [12] Y. Jiao and N. Chawla, *J. Appl. Phys.* **115**, 093511 (2014).
- [13] K. M. Gerke, M. V. Karsanina, R. V. Vasilyev, and D. Mallants, *Europhys. Lett.* **106**, 66002 (2014).
- [14] K. M. Gerke and M. V. Karsanina, *Eur. J. Soil Sci.* **72**, 527 (2021).
- [15] K. M. Gerke, M. V. Karsanina, and D. Mallants, *Sci. Rep.* **5**, 15880 (2015).
- [16] J. Havelka, A. Kučerová, and J. Sýkora, *Comput. Mater. Sci.* **122**, 102 (2016).
- [17] S. Chen, H. Li, and Y. Jiao, *Phys. Rev. E* **92**, 023301 (2015).
- [18] Y. Xu, P.-E. Chen, H. Li, W. Xu, Y. Ren, W. Shan, and Y. Jiao, *J. Appl. Phys.* **131**, 115104 (2022).
- [19] X. Miao, K. M. Gerke, and T. O. Sizonenko, *Adv. Water Resour.* **105**, 162 (2017).
- [20] S. Kamrava, P. Tahmasebi, and M. Sahimi, *Transp. Porous Media* **131**, 427 (2020).
- [21] M. Röding, Z. Ma, and S. Torquato, *Sci. Rep.* **10**, 1 (2020).
- [22] P. Adler, C. G. Jacquin, and J. Quiblier, *Int. J. Multiphase Flow* **16**, 691 (1990).
- [23] C. L. Y. Yeong and S. Torquato, *Phys. Rev. E* **57**, 495 (1998).
- [24] K. M. Gerke and M. V. Karsanina, *Europhys. Lett.* **111**, 56002 (2015).
- [25] P. Tahmasebi and M. Sahimi, *Phys. Rev. Lett.* **110**, 078002 (2013).
- [26] M. V. Karsanina and K. M. Gerke, *Phys. Rev. Lett.* **121**, 265501 (2018).
- [27] S. Chen, A. Kirubanandham, N. Chawla, and Y. Jiao, *Metall. Mater. Trans. A* **47**, 1440 (2016).
- [28] M. V. Karsanina, K. M. Gerke, E. B. Skvortsova, A. L. Ivanov, and D. Mallants, *Geoderma* **314**, 138 (2018).
- [29] P. Čapek, V. Hejtmánek, L. Brabec, A. Zikánová, and M. Kočířík, *Transp. Porous Media* **76**, 179 (2009).
- [30] Y. Jiao, F. Stillinger, and S. Torquato, *Proc. Natl. Acad. Sci. U.S.A.* **106**, 17634 (2009).
- [31] C. J. Gommès, Y. Jiao, and S. Torquato, *Phys. Rev. Lett.* **108**, 080601 (2012).
- [32] C. J. Gommès, Y. Jiao, and S. Torquato, *Phys. Rev. E* **85**, 051140 (2012).
- [33] M. Skolnick and S. Torquato, *Phys. Rev. E* **104**, 045306 (2021).
- [34] P. Debye, H. Anderson Jr., and H. Brumberger, *J. Appl. Phys.* **28**, 679 (1957).
- [35] H. Li, P.-E. Chen, and Y. Jiao, *Transp. Porous Media* **125**, 5 (2018).
- [36] C. J. Gommès, Y. Jiao, A. P. Roberts, and D. Jeulin, *J. Appl. Crystallogr.* **53**, 127 (2020).
- [37] S. Dietrich and A. Haase, *Phys. Rep.* **260**, 1 (1995).
- [38] Z. Ma and S. Torquato, *Phys. Rev. E* **98**, 013307 (2018).
- [39] K. M. Gerke, E. V. Korostilev, K. A. Romanenko, and M. V. Karsanina, *Geoderma* **383**, 114739 (2021).
- [40] D. Wildenschild and A. P. Sheppard, *Adv. Water Resour.* **51**, 217 (2013).
- [41] T. Bultreys, W. De Boever, and V. Cnudde, *Earth Sci. Rev.* **155**, 93 (2016).
- [42] E. V. Lavrukhin, K. M. Gerke, K. A. Romanenko, K. N. Abrosimov, and M. V. Karsanina, *Soil Tillage Res.* **209**, 104942 (2021).
- [43] K. M. Gerke, T. O. Sizonenko, M. V. Karsanina, E. V. Lavrukhin, V. V. Abashkin, and D. V. Korost, *Adv. Water Resour.* **140**, 103576 (2020).
- [44] A. S. Zubov, D. A. Murygin, and K. M. Gerke, *Phys. Rev. E* **106**, 055304 (2022).
- [45] F. Frank, C. Liu, A. Scanziani, F. O. Alpak, and B. Riviere, *J. Colloid Interface Sci.* **523**, 282 (2018).
- [46] K. Gerke, E. Skvortsova, and D. Korost, *Eurasian Soil Sci.* **45**, 700 (2012).
- [47] S. Torquato, *Random Heterogeneous Materials: Microstructure and Macroscopic Properties* (Springer-Verlag, New York, 2002), p. 703.
- [48] S. Torquato, *Phys. Rev. E* **94**, 022122 (2016).
- [49] L. Debnath and P. Mikusinski, *Introduction to Hilbert Spaces with Applications*, 3rd ed. (Academic Press, San Diego, CA, 2005).
- [50] W. G. Bickley, *Q. J. Mech. Appl. Math.* **1**, 35 (1948).
- [51] J. M. S. Prewitt, *Picture Processing and Psychopictorics* (Academic Press, New York, 1970), Vol. 3, pp. 58–71.
- [52] S. Ando, *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 252 (2000).

- [53] M. Salzer, A. Spettl, O. Stenzel, J.-H. Smått, M. Lindén, I. Manke, and V. Schmidt, *Mater. Charact.* **69**, 115 (2012).
- [54] P. Iassonov, T. Gebrenegus, and M. Tuller, *Water Resour. Res.* **45**, W09415 (2009).
- [55] K. Abrosimov, K. Gerke, I. Semenov, and D. Korost, *Eurasian Soil Sci.* **54**, 560 (2021).
- [56] N. M. Evstigneev and V. Postnicov, Ma-torquato's algorithm for computation of surface correlation functions (2d case), <https://gist.github.com/shamazmzum/47a3d2dda58ae9d6481d63a35e578523> (2022).
- [57] A. Khlyupin and O. Y. Dinariev, *Tech. Phys.* **60**, 805 (2015).
- [58] T. Aslyamov, A. Khlyupin, V. Pletneva, and I. Akhatov, *J. Phys. Chem. C* **123**, 28707 (2019).
- [59] A. Khlyupin, *J. Phys.: Conf. Ser.* **774**, 012024 (2016).
- [60] J. Feder, *Fractals* (Springer, Berlin, 1988).
- [61] A. Moctezuma-Berthier, O. Vizika, and P. Adler, *Transp. Porous Media* **49**, 313 (2002).
- [62] J. R. Godinho, K. M. Gerke, A. G. Stack, and P. D. Lee, *Sci. Rep.* **6**, 33086 (2016).
- [63] C. Noiriel and C. Soulaïne, *Transp. Porous Media* **140**, 181 (2021).
- [64] D. Prokhorov, V. Lisitsa, T. Khachkova, Y. Bazaikin, and Y. Yang, *J. Comput. Sci.* **58**, 101550 (2022).
- [65] P.-E. Chen, R. Raghavan, Y. Zheng, H. Li, K. Ankit, and Y. Jiao, *Phys. Rev. E* **105**, 025306 (2022).
- [66] M. G. Rozman and M. Utz, *Phys. Rev. E* **63**, 066701 (2001).
- [67] H. Malmir, M. Sahimi, and Y. Jiao, *Phys. Rev. E* **98**, 063317 (2018).
- [68] H. Chen, X. He, Q. Teng, R. E. Sheriff, J. Feng, and S. Xiong, *Phys. Rev. E* **101**, 023305 (2020).
- [69] N. Janssens, M. Huysmans, and R. Swennen, *Materials* **13**, 1397 (2020).
- [70] S. Karimpouli and R. Kadyrov, *J. Pet. Sci. Eng.* **216**, 110833 (2022).
- [71] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevE.107.065306> for Jupiter Notebook with code examples demonstrating computations of surface correlation functions.
- [72] www.porennetwork.com.